UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/618,882 | 07/11/2003 | Chih-Wei Chen | LA-7196-113.US | 6816 |

167 7590 01/26/2006

FULBRIGHT AND JAWORSKI LLP
555 S. FLOWER STREET, 41ST FLOOR
LOS ANGELES, CA 90071

| EXAMINER | |
|---|---|
| PATEL, KAUSHIKKUMAR M | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2188 | |

DATE MAILED: 01/26/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 10/618,882 | CHEN, CHIH-WEI | |
| | Examiner | Art Unit | |
| | Kaushikkumar Patel | 2188 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *03 November 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-10* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-10* is/are rejected.

7)☒ Claim(s) *3-5* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *11 July 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

## DETAILED ACTION

### *Response to Amendment*

1. This Office action is in response to applicant's communication filed on November 3, 2005. The applicant's remarks and amendments to the claims were considered with the results that follow.

2. In response to last office action claim 1 is amended. No claims have been added or canceled. As a result, claims 1-10 remain pending in this application.

### *Response to Arguments*

3. Applicant's arguments filed November 3, 2005 have been fully considered but they are not persuasive.

4. Applicant argues that teaching of Kim et al. (US 2003/0023811 A1) is posterior processing. However Kim mentions that the method can be used for new (fig. 10, paragraph [0120] as well as for existing volumes (fig. 11, paragraph [0122]). Kim also provides Admin GUI and command line interfaces (fig. 1, item 20), which provides the user interface for user (administrator) to enter parameter file (user specified settings) to create the logical volumes.

5. In response to above-mentioned arguments, examiner respectfully maintains the rejections of the claims and repeated herein after as follows.

### *Claim Objections*

6. Claims 3, 4, and 5 are objected to because of the following informalities:

Claims 3 and 4, line 2, "step (1)" should be "step (2)" according to new amendment filed on November 3, 2005.

Claim 5, line 2, "step (3)" should be ""step (4)".

Appropriate correction is required.

### *Claim Rejections - 35 USC § 102*

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

8. Claims 1-10 are rejected under 35 U.S.C. 102(b) as being anticipated by Kim et al (US 2003/0023811 A1).

Claim 1 is taught by Kim as:

A multi-volume disk array management method for use on a multi-disk storage unit having a number of disks for the purpose of allowing the multi-disk storage unit to provide at least two logical volumes for storing data in the logical volumes with at least two levels of fault tolerance (see figure 2, items 51, 52, 53 and 54 are multi-disks with items 61, 62, 63, and 64 are multi-volumes with multiple fault tolerance levels);

the multi-volume disk array management method comprising:

(1) receiving user specified settings related to management of the overall storage space or the multi-disk storage unit (Kim teaches an Admin GUI and command line interface in fig.1, item 20, and uses parameter file (user specified settings) to create new volume, fig 10); (according to Microsoft Computer Dictionary, Fifth edition, page 111, command line interface is a form of interface between user and operating system in which user types commands and page 239, GUI graphical user interface visual interface to manage computer environment);

(2) logically dividing the storage space of each of the disks in the multi-disk storage unit into a number of partitions (see figure 2 and paragraph [0071], lines 3-4);

(3) organizing at least two selected subgroups of partitions in the disks of the multi-disk storage unit into at least two logical volumes (see figure 2, items 61, 62, 63 and 64, paragraphs [0071,0078, and 0079], lines 1-2); and

(4) setting the storage property of each of the logical volumes in the multi-disk storage unit into a user-specified level of fault tolerance (see figure 1, item 20 for admin GUI and command line interface which specifies user interface and paragraphs [0072-0075] for different levels of fault tolerances).

9. Claim 2 is taught by Kim as:

The multi-volume disk array management method of claim 1, wherein the multi-disks storage unit is RAID-compliant storage unit (Kim teaches different levels of RAID levels for storage, see paragraphs [0072-0075]).

10. Claim 3 is taught by Kim as:

The multi-volume disk array management method of claim 1, wherein said step (2), FDISK disk management utility is utilized to logically divide the storage space of each of the disks in the multi-disk storage unit into a number of partitions (see paragraph [0071], lines 5-6).

11. Claim 4 is taught by Kim as:

The multi-volume disk array management method of claim 1, wherein said step (2), all the partitions are set to be equal in size (see paragraph [0111], line 3, Kim talking about the partitions are equal in size).

12. Claim 5 is taught by Kim as:

The multi-volume disk array management method of claim 2, wherein said step (4), each user-specified level of fault tolerance is a RAID-compliant level of tolerance (see paragraphs [0072-0074], which teaches different levels of RAID-compliant fault tolerances).

13. Claim 6 is taught by Kim as:

A multi-volume disk array management system for use on a multi-disk storage unit having a number of disks for the purpose of allowing the multi-disk storage unit to provide at least two logical volumes for storing data in the logical volumes with at least two levels of fault tolerance (see figure 2, items 51, 52, 53 and 54 are multi-disks with items 61, 62, 63, and 64 are multi-volumes with multiple fault tolerance levels);

the multi-volume disk array management system comprising:

a user interface for receiving user-specified settings related to the management of the overall storage space of the multi-disk storage unit (see figure 1, item 20, paragraph [0069], lines 3-6);

a storage-space partitioning module, which is capable of logically dividing the storage space of each of the disks in the multi-disk storage unit into a number of partitions based on the user-specified settings from the user interface (see, paragraph [0076] in which Kim teaches three virtualizations or modules and paragraph [0077] teaches a partitioning module);

a logical-volume organizing module, which is capable of organizing at least two selected subgroups of partitions in the disks of the multi-disk storage unit into at least two logical volumes based on the user-specified settings from the user interface (see paragraph [0079], which teaches a logical volume creation module);

a storage-property setting module, which is capable of setting the storage property of each of the logical volumes in the multi-disk storage unit into a user-specified level of fault tolerance based on the user-specified settings from the user interface (see figure 1, item 20 for admin GUI and command line interface which specifies user interface and paragraph [0080-0081]).

14. Claim 7 is taught by Kim as:

The multi-volume disk array management system of claim 6, wherein the multi-disks storage unit is RAID-compliant storage unit (Kim teaches different levels of RAID levels for storage, see paragraphs [0072-0075]).

15. Claim 8 is taught by Kim as:

The multi-volume disk array management system of claim 6, wherein said step (2), FDISK disk management utility is utilized to logically divide the storage space of each of the disks in the multi-disk storage unit into a number of partitions (see paragraph [0071], lines 5-6).

16. Claim 9 is taught by Kim as:

The multi-volume disk array management system of claim 6, wherein said step (2), all the partitions are set to be equal in size (see paragraph [0111], line 3, Kim talking about the partitions are equal in size).

17. Claim 10 is taught by Kim as:

The multi-volume disk array management system of claim 7, wherein the user-specified level of fault tolerance is based on the RAID-compliant levels of tolerance (see paragraphs [0072-0074], which teaches different levels of RAID-compliant fault tolerances).

***Conclusion***

18. The prior arts made of the record and not relied upon are considered pertinent to applicant's discloser. Gentry et al (5,568,629) teach a method for partitioning a disk array into multiple logical storage units with different schemes (RAID levels) for storing data. Stallmo (6,15,854) teaches the redundant array storage system multiple logical volumes, which can be set at different redundancy groups.

19. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kaushikkumar Patel whose telephone number is 571-272-5536. The examiner can normally be reached on 8.00 am - 4.30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Mano Padmanabhan can be reached on 571-272-4210. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Kaushikkumar Patel
Examiner
Art Unit 2188

kmp

MANO PADMANABHAN
SUPERVISORY PATENT EXAMINER

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399



Library of Congress Cataloging-in-Publication Data
Microsoft Computer Dictionary.--5th ed..
p. ; cm.
ISBN 0-7356-1495-4
1. Computers--Dictionaries. 2. Microcomputers--Dictionaries.

AQ76.5. M52267 2002
004'.03--dc21 200219714

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 QWT 7 6 5 4 3 2

Distributed in Canada by Penguin Books Canada Limited.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office or contact Microsoft Press International directly at fax (425) 936-7329. Visit our Web site at www.microsoft.com/mspress. Send comments to *mspinput@microsoft.com*.



**Acquisitions Editor:** Alex Blanton
**Project Editor:** Sandra Haynes

Body Part No. X08-41929

graph is weighted if each edge has some value associated with it. *See also* node (definition 3), tree. **2.** *See* chart.

**Graphical Device Interface** *n.* *See* GDI.

**graphical interface** *n.* *See* graphical user interface.

**Graphical Kernel System** *n.* A computer graphics standard, recognized by ANSI and ISO, that specifies methods of describing, manipulating, storing, and transferring graphical images. It functions at the application level rather than the hardware level and deals with logical workstations (combinations of input and output devices such as keyboard, mouse, and monitor) rather than with individual devices. Graphical Kernel System was developed in 1978 to handle two-dimensional graphics; the later modification, GKS-3D, extended the standard to three-dimensional graphics. *Acronym:* GKS. *See also* ANSI, ISO.

**graphical user interface** *n.* A visual computer environment that represents programs, files, and options with graphical images, such as icons, menus, and dialog boxes, on the screen. The user can select and activate these options by pointing and clicking with a mouse or, often, with the keyboard. A particular item (such as a scroll bar) works the same way for the user in all applications, because the graphical user interface provides standard software routines to handle these elements and report the user's actions (such as a mouse click on a particular icon or at a particular location in text, or a key press); applications call these routines with specific parameters rather than attempting to reproduce them from scratch. *Acronym:* GUI.

**graphic character** *n.* Any character that is represented by a visible symbol, such as an ASCII character. A graphic character is not the same as a graphics character. *Compare* graphics character.

**graphic limits** *n.* On a computer screen, the boundary of a graphical image in a graphics software program, including all the area enclosed within the graphic. In some graphics environments the limits of a graphic consist of the smallest rectangle that can completely enclose it, called its *bounding rectangle* or *bounding box*.

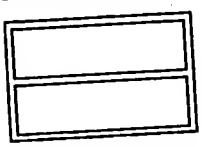
**graphics accelerator** *n.* A video adapter that contains a graphics coprocessor. A graphics accelerator can update the video display much more quickly than the CPU can, and it frees the CPU for other tasks. A graphics accelerator is a necessity for modern software such as graphical user interfaces and multimedia applications. *See also* graphics coprocessor, video adapter.



**graphics adapter** *n.* A video adapter capable of displaying graphics as well as alphanumeric characters. Almost all video adapters in common use today are graphics adapters.

**graphics card** *n.* *See* video adapter.

**graphics character** *n.* A character that can be combined with others to create simple graphics, such as lines, boxes, and shaded or solid blocks. See the illustration. *Compare* graphic character.



***Graphics character.*** *Box built up from line graphics characters.*

**graphics controller** *n.* The part of the EGA and VGA video adapters that allows the computer to access the video buffer. *See also* EGA, VGA.

**graphics coprocessor** *n.* A specialized microprocessor, included in some video adapters, that can generate graphical images such as lines and filled areas in response to instructions from the CPU, freeing the CPU for other work.

**graphics data structure** *n.* A data structure that is designed specifically for representing one or more elements of a graphical image.

**graphics engine** *n.* **1.** A display adapter that handles high-speed graphics-related processing, freeing the CPU for other tasks. *Also called:* graphics accelerator, video accelerator. **2.** Software that, based on commands from an application, sends instructions for creating graphic images to the hardware that actually creates the images. Examples are Macintosh QuickDraw and Windows Graphics Device Interface (GDI).

**graphics export component** *n.* A technology developed by Apple for creating, editing, publishing, and viewing multimedia content. The graphics export component provides an application programming interface that enables a QuickTime player to export still images into a variety of file formats.

**graphics import component** *n.* A technology developed by Apple for creating, editing, publishing, and viewing multimedia content. The graphics import component provides an application programming interface that enables a QuickTime player to import still images from a variety of file formats.

A permutation is a grouping of elements taken from a er set with regard to the order of the elements. For mple, in making permutations of two objects from the e set of four objects, there would be four candidates to ose from for the first selection (A), and three left over hoose from for the second selection (B), or 12 permuta- s in all: AB, AC, AD, BA, BC, BD, CA, CB, CD, DA, , DC. *See also* combinatorial explosion.

**M callable wrapper** *n.* A proxy object generated by untime so that existing COM applications can use naged classes, including .NET Framework classes, nsparently. *Acronym:* CCW.

**OMDEX** *n.* Any of a series of annual computer trade ows operated by Softbank COMDEX, Inc. One of these ows takes place in Las Vegas each November and is the gest computer trade show in the United States.

**mité Consultatif International Télégraphique et léphonique** *n. See* CCITT.

**mma-delimited file** *n.* A data file consisting of fields d records, stored as text, in which the fields are separated m each other by commas. Use of comma-delimited files ows communication between database systems that use fferent formats. If the data in a field contains a comma, field is further surrounded with quotation marks.

**mmand** *n.* An instruction to a computer program that, hen issued by the user, causes an action to be carried out. ommands are usually either typed at the keyboard or osen from a menu.

**mmand buffer** *n.* An area in memory in which com- ands entered by the user are kept. A command buffer can able the user to repeat commands without retyping them mpletely, edit past commands to change some argument correct a mistake, undo commands, or obtain a list of ast commands. *See also* history, template (definition 4).

**ommand button** *n.* A control shaped like a pushbutton a dialog box in a graphical user interface. By clicking a ommand button, the user causes the computer to perform me action, such as opening a file that has just been elected using the other controls in the dialog box.

**COMMAND.COM** *n.* The command interpreter for MS- OS. *See also* command interpreter.

**command-driven** *adj.* Accepting commands in the form of code words or letters, which the user must learn. *Compare* menu-driven.

**command-driven system** *n.* A system in which the user initiates operations by a command entered from the console. *Compare* graphical user interface.

**command interpreter** *n.* A program, usually part of the operating system, that accepts typed commands from the keyboard and performs tasks as directed. The command interpreter is responsible for loading applications and directing the flow of information between applications. In OS/2 and MS-DOS, the command interpreter also handles simple functions, such as moving and copying files and displaying disk directory information. *See also* shell[1].

**Command key** *n.* On the original Macintosh keyboard, a key labeled with the special symbol, sometimes called the propeller or puppy foot. This key is found on one or both sides of the Spacebar, depending on the version of the Apple keyboard. The key serves some of the same functions as the Control key on IBM keyboards. *See also* Control key.

**command language** *n.* The set of keywords and expressions that are accepted as valid by the command interpreter. *See also* command interpreter.

**command line** *n.* A string of text written in the command language and passed to the command interpreter for execution. *See also* command language.

**command-line interface** *n.* A form of interface between the operating system and the user in which the user types commands, using a special command language. Although systems with command-line interfaces are usually considered more difficult to learn and use than those with graphical interfaces, command-based systems are usually programmable; this gives them flexibility unavailable in graphics-based systems that do not have a programming interface. *Compare* graphical user interface.

**command mode** *n.* A mode of operation in which a program waits for a command to be issued. *Compare* edit mode, insert mode.

**command processing** *n. See* command-driven system.

**command processor** *n. See* command interpreter.

**command prompt window** *n.* A window displayed on the desktop used to interface with the MS-DOS operating